

# Documentação das APIs dos Microsserviços

## Microsserviço: Autenticação (Auth)

Descrição: Gerencia o registro e login de usuários e gera tokens JWT para autenticação.

### Endpoints:

#### 1.Registro de Usuário

- Método: *POST*
- URL: <http://localhost:3002/auth/register>
- Body Exemplo:

```
{  
  "name": "João Silva",  
  "email": "joao@example.com",  
  "password": "senha123"  
}
```

- Descrição: Registra um novo usuário no sistema.

#### 2.Login de Usuário

- Método: *POST*
- URL: <http://localhost:3002/auth/login>
- Body Exemplo:

```
{  
  "email": "joao@example.com",  
  "password": "senha123"  
}
```

- Resposta:

```
{  
  "token": "seu-token-jwt-aqui"  
}
```

- Descrição: Realiza o login e retorna um token JWT. Este token deve ser usado para acessar rotas protegidas.

### 3.Verificar Token

- Método: *GET*
- URL: <http://localhost:3002/auth/verify>
- Headers:



- Descrição: Verificar se o token JWT é válido.

### Microserviço: Gerenciamento de Usuários (Professores e Alunos)

Descrição: Permite criar, listar, buscar, atualizar e excluir usuários.

Algumas operações podem ser protegidas via token JWT, dependendo da configuração de middleware.

#### Endpoints:

##### 1.Criar Usuário

- Método: *POST*
- URL: <http://localhost:3003/users>
- Body Exemplo:

```
{
  "name": "Maria Souza",
  "email": "maria@example.com",
  "role": "student"
}
```

- Descrição: Cria um novo usuário.

##### 2.Listar todos Usuários

- Método: *GET*
- URL: <http://localhost:3003/users>

##### 3.Buscar Usuário por ID

- Método: *GET*
- URL: <http://localhost:3003/users/:id>

#### 4. Atualizar Usuário

- Método: *PUT*
- URL: <http://localhost:3003/users/:id>
- Body Exemplo:

```
{  
  "name": "Maria Souza Atualizada",  
  "email": "mariaatualizada@example.com",  
  "role": "teacher"  
}
```

#### 5. Deletar Usuário

- Método: *DELETE*
- URL: <http://localhost:3003/users/:id>

#### Microsserviço: Gerenciamento de Posts

Descrição: Permite criar, listar, buscar, atualizar e excluir posts.

As operações de criar, editar e excluir requerem autenticação (token JWT).

##### 1. Criar Posts (protegido)

- Método: *POST*
- URL: <http://localhost:3001/posts>
- Headers:

<input checked="" type="checkbox"/> Authorization	Bearer <seu-token-jwt>
---	------------------------

- Body Exemplo:

```
{  
  "title": "Título do Post",  
  "content": "Conteúdo do Post",  
  "authorId": 1  
}
```

##### 2. Listar todos os Posts

- Método: *GET*
- URL: <http://localhost:3001/posts>

### 3. Buscar Post por ID

- Método: *GET*
- URL: <http://localhost:3001/posts/:id>

### 4. Atualizar Post (protegido)

- Método: *PUT*
- URL: <http://localhost:3001/posts/:id>
- Headers:

<input checked="" type="checkbox"/> Authorization	Bearer <seu-token-jwt>
---	------------------------

- Body Exemplo:

```
{
  ... "title": "Título Atualizado",
  ... "content": "Conteúdo Atualizado"
}
```

### 5. Deletar Post (protegido)

- Método: *DELETE*
- URL: <http://localhost:3001/posts/:id>
- Headers:

<input checked="" type="checkbox"/> Authorization	Bearer <seu-token-jwt>
---	------------------------